**Software Implementation and Testing Document**

**For**

**Group 4: Retro-Arcade**

Version 3.0

**Authors**:

Andrew L

Joseph B

Mackenzie P

Michael T

Seth P

# Programming Languages (5 points)

Python3 – used for GUI and game design.

MySQL – Backend database to keep track of user profiles, high scores, and user stats of games played.

# Platforms, APIs, Databases, and other technologies used (5 points)

PySimpleGUI – used to create GUI in Python

Pygames – used to develop the actual games

SQLite – database API using MariaDB for database management

Flask – API used to create virtual local web server to show highscores pages (in HTML)

Cryptography – API used to have Flask include SSL 2.0

# Execution-based Functional Testing (10 points)

Flappy Bird correctly tells the user prompts on starting the game as well as when the user has lost. The score for the user is displayed to the screen. For Space Invaders, two kinks need to be ironed out, we have an issue where aliens will replace others after destroyed and the ship’s rockets do not disappear on new level.

Flask server runs properly, successfully tested CRUD functions for Pong. Dummy Data Pong leaderboard works. Flappy Bird leaderboard not yet tested.

# Execution-based Non-Functional Testing (10 points)

Our games have not crashed after letting them run for a long period of time.

Basic CSS style sheet works properly.

# Non-Execution-based Testing (10 points)

Space Invaders’ code is clean and legible. Flappy Bird needs to be trimmed of some unnecessary commented code.

Snake’s code is clean and legible, just need to add some functionality comments.

Database tables are partially normalized, webpages have no broken link, are easy to navigate.